

ARQUITECTURAS ADAPTATIVAS E COOPERATIVAS PARA SISTEMAS DE FABRICO DISTRIBUÍDOS

Paulo Leitão *, Francisco Restivo **

* Instituto Politécnico de Bragança, Quinta S^{ta} Apolónia, Apartado 134, P-5301-857 Bragança, pleitao@ipb.pt

** Faculdade de Engenharia da Universidade do Porto, Rua Dr. Roberto Frias, P-4200-465 Porto, fjr@fe.up.pt

Resumo

A pressão competitiva actual, é ditada por um mercado aberto e global, e no qual as empresas são confrontadas com exigências cada vez mais fortes. A adopção de novos conceitos e paradigmas de fabrico, combinados com a implementação de novas tecnologias é a resposta a estes requisitos e a solução para o incremento da competitividade. Arquitecturas de controlo distribuídas, cooperativas, auto-organizativas e heterogéneas são prometedoras neste contexto.

A arquitectura apresentada, pretende solucionar de forma eficaz os requisitos dos sistemas de fabrico emergentes, sendo baseada na tecnologia de multi-agentes, e implementando alguns conceitos associados aos sistemas de fabrico holónico e biónico, suportando uma nova abordagem ao controlo, que inclui as fases operacional e de reengenharia do ciclo de vida de uma empresa.

1. INTRODUÇÃO

A competição global e aberta entre empresas e as novas estruturas organizativas de negócio conduzem à necessidade de novos sistemas que executem o controlo e monitorização de sistemas distribuídos, através da integração de sistemas de informação e características de auto-organização, de forma a adaptarem-se rapidamente às mudanças.

Este artigo apresenta uma breve análise da evolução dos paradigmas de fabrico, desde os tradicionais até aos ambientes de fabrico distribuídos, assim como os requisitos e problemas associados aos sistemas de fabrico emergentes. São apresentados alguns dos trabalhos de investigação mais relevantes na área das arquitecturas de controlo para sistemas de fabrico, tanto as abordagens tradicionais como as avançadas, assim

como os novos paradigmas para sistemas de fabrico distribuídos.

De forma a cumprir os requisitos apresentados pelos sistemas de fabrico distribuídos e cooperativos, é apresentada uma arquitectura baseada em agentes que implementa alguns conceitos associados aos sistemas de fabrico holónicos e biónicos, e que utiliza os sistemas multi-agentes como plataforma.

Outro ponto importante neste estudo é a introdução de uma nova abordagem ao controlo, que agrupa as fases operacionais e de reengenharia do ciclo de vida de uma empresa.

2. AMBIENTES DE FABRICO DISTRIBUÍDOS

2.1. Evolução dos paradigmas de fabrico

Durante vários anos, o conceito de produção em massa, caracterizado pela produção do mesmo produto em larga escala, foi largamente aceite e implementado. Actualmente, este paradigma já não responde satisfatoriamente aos desafios do mercado, que exige sistemas produtivos modernos, globais e dinâmicos, e é incapaz de tratar variações no tipo de produto.

Nos anos 80, as empresas Japonesas introduziram o paradigma *Lean Manufacturing*, que é uma extensão do *Just in Time*, que consiste em ter o material devido no local correcto e na hora certa, eliminando *stocks*, e utilizando sistemas de controlo e escalonamento muito simples [1]. O *Lean Manufacturing* pretende encurtar o tempo entre o pedido de encomenda e o envio do produto ao cliente, através da eliminação do desperdício, e produzindo produtos com menor tempo de projecto, menor inventário, menor número de defeitos, redução dos setups, etc. O *Agile Manufacturing* é um paradigma semelhante, que consiste na capacidade de adaptação rápida e

proveitosa às mudanças contínuas e inesperadas no ambiente de fabrico. Neste paradigma, é importante considerar que os factores humanos e o conhecimento organizativo tem a mesma importância que a tecnologia utilizada.

O paradigma CIM (*Computer Integrated Manufacturing*), popular nos anos 80, consiste na integração das actividades de uma empresa, através do uso de tecnologias de informação, tais como base de dados e redes de comunicação, que permitem a troca e partilha de dados entre várias unidades da empresa. No início, a integração apenas envolveu as actividades de engenharia e de produção, que evoluiu para a integração com os clientes e fornecedores, conduzindo ao conceito de gestão da cadeia produtiva (*Supply Chain Management*). As principais vantagens do conceito CIM são o incremento de produtividade, flexibilidade, qualidade e redução do WIP (*Work in Progress*) [2; 3]. No entanto, a implementação do conceito CIM não obteve os resultados esperados, devido principalmente a aspectos tecnológicos, económicos e sociais.

A gestão da cadeia produtiva é um conceito associado aos sistemas de fabrico distribuídos, que envolve a gestão do fluxo de materiais, de informação, e financeiro, numa rede que engloba fornecedores, fabricantes, distribuidores e clientes. O paradigma de empresa virtual pode ser definido como uma aliança temporária de empresas que se juntam para partilhar conhecimento e recursos, de forma a melhor responder às oportunidades de negócio, sendo essa cooperação suportada por redes de computadores [4]. As empresas estendidas são um caso particular das empresas virtuais, onde uma empresa assume um papel preponderante na rede de organizações.

Actualmente, a tendência é que cada produto possua diversos modelos, cada um dos quais altamente customizado de acordo com os requisitos dos clientes, o que implica a necessidade de utilizar produção do tipo *job-shop* e em alguns casos *one-of-a-kind*, e a necessidade de uma adaptação ágil às mudanças. Desta forma, o fabrico emergiu da era do fabrico em massa para a era da customização em massa, sendo as empresas obrigadas a utilizar novas tecnologias e novos conceitos de fabrico, combinando-os, para evitar o risco de se tornarem menos competitivos ou obsoletos.

2.2. Requisitos dos sistemas de fabrico emergentes

As novas formas organizativas de negocio apresentam requisitos e problemas, que incrementam a complexidade no desenvolvimento de aplicações de controlo. Os principais requisitos dos sistemas de fabrico emergentes são [1; 5]: integração de empresa, arquitecturas organizativas distribuídas, ambientes heterogéneos, integração de humanos, cooperação, estruturas abertas e dinâmicas, estrutura organizativa dinâmica e tolerância a falhas.

O desenvolvimento de aplicações de fabrico que cumpram os requisitos e suportem as novas estruturas

organizativas, apresentam problemas importantes que ainda não estão completamente solucionados, tais como a troca de dados, o planeamento e escalonamento descentralizado e técnicas de re-organização [1]. A troca de dados, está relacionada com o facto de, num ambiente distribuído, cada empresa representar e interpretar a informação de forma diferente. Os protocolos EDI (*Electronic Data Interchange*) e STEP (*Standard for the Exchange of Product model data*) não solucionam completamente o problema, e não cobrem outras domínios de informação, tais como a informação relacionada com a qualidade e a informação relacionada com o processo. Recentemente, o XML (*eXtensible Markup Language*) tem sido apontado como o standard para a troca de dados. No entanto, apesar de facilitar os aspectos de sintaxe na troca de dados, o estabelecimento de ontologias comuns continua a ser um desafio difícil. Num ambiente distribuído, é necessário ligar os sistemas de planeamento e escalonamento de uma empresa com os sistemas dos seus clientes e fornecedores. Esta descentralização requer a sincronização dos sistemas de planeamento e escalonamento local e supervisão baseada numa plataforma descentralizada para suporte à optimização global. Finalmente, a representação de técnicas de reorganização e as responsabilidades associadas ao início do processo de reorganização é uma tarefa complexa, que requer investigação adicional, de forma a definir um modelo standard para a representação dessas técnicas.

3. ARQUITECTURAS DE CONTROLO EXISTENTES

Na comunidade de investigação, diversas arquitecturas de controlo foram propostas de forma a cumprir os requisitos dos sistemas de fabrico emergentes, algumas usando abordagens clássicas (baseadas essencialmente nas arquitecturas de controlo centralizado e hierárquico), e outras usando novas e avançadas abordagens (baseadas em arquitecturas mais heterarquicas e usando sistemas de multi-agentes).

3.1. Arquitecturas de controlo clássicas

O projecto COSIMA (*Control Systems for Integrated Manufacturing*) [6], desenvolveu uma arquitectura hierárquica para os níveis de planta fabril e de célula, e consiste em cinco módulos funcionais, agrupados no conceito PAC (*Production Activity Control*): escalonador, despacho, monitor, produtor e manipulador. A arquitectura CHAMP (*Chalmers Architecture and Methodology for Flexible Production*) foi elaborada em cima das experiências da arquitectura PAC e das suas extensões, PAC+ e PAC++. As principais características desta arquitectura são a separação da informação relacionada com o produto e com o recurso, e a separação das funções genéricas e funções específicas dos produtos e recursos correntemente em uso [7]. Similar a estas arquitecturas, é o controlador de célula de fabrico do Centro de CIM do Porto, que utiliza uma arquitectura hierárquica

modificada. Esta arquitectura especifica um conjunto de módulos, cujo cérebro é o módulo Gestor e no qual cada dispositivo físico possui um controlador de dispositivo, que é customizado para o dispositivo em questão. A interface entre o controlador de célula e os dispositivos físicos é implementado utilizando o protocolo de comunicação MMS (*Manufacturing Message Specification*) [8].

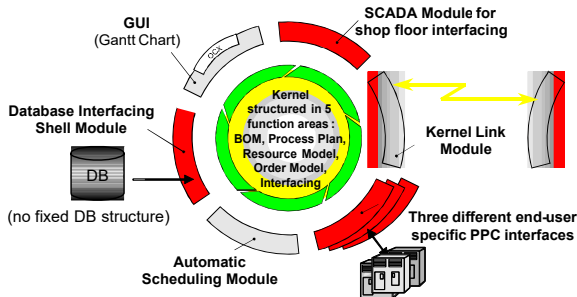


Figura 1 – Arquitectura MOSCOT

A arquitectura MOSCOT (*Modular Shop Control Toolkit for Flexible Manufacturing*) é caracterizada por duas componentes básicas: o *kernel*, que contém os módulos comuns a todas as aplicações de controlo da planta fabril, e *shells*, que são desenvolvidas ou customizadas de acordo com as especificações de cada aplicação [9]. Outra inovação associada a esta arquitectura é a SCAPI (*Shop Control Application Program Interface*), que actua como um sistema operativo para aplicações de controlo da planta fabril.

3.2 Paradigmas de controlo para fabrico distribuído

Os sistemas de controlo de fabrico tradicionais apresentam baixa capacidade de adaptação e reacção às mudanças dinâmicas do seu ambiente, tais como reacção a distúrbios e reacção às mudanças de mercado. Os novos paradigmas devem apresentar a possibilidade de responder rapidamente e correctamente a mudanças externas, e diferem das abordagens tradicionais devido à sua capacidade de adaptação a mudanças sem intervenção externa.

De forma a atingir as características e comportamentos referidos, existem várias teorias, que apresentam conceitos e características similares, mas que diferem na sua origem: matemática (teoria do caos) para a fabrica fractal [10], biologia para os sistemas de fabrico biónicos (BMS) [11] e organização social para os sistemas de fabrico holónicos (HMS) [12].

Uma apresentação destas teorias pode ser encontrada em [1] e a comparação entre os vários paradigmas pode ser encontrada em [13].

3.3. Arquitecturas de controlo avançadas

As arquitecturas de controlo para os sistemas de fabrico emergentes utilizam a tecnologia multi-agente para suportar o desenvolvimento de arquitecturas adaptativas e autónomas. A arquitectura PROSA (*Product-Resource-Order-Staff Architecture*) é uma arquitectura holónica de referência, baseada em três

tipos de holons básicos: produto, ordem e recurso [12]. Adicionalmente, existem holons staff cuja missão é assistir e aconselhar os holons básicos. O projecto MetaMorph é uma arquitectura multi-agente cujo objectivo é integrar as actividades de uma empresa de fabrico com as dos seus fornecedores, clientes e parceiros num ambiente distribuído [14]. O projecto propõe uma arquitectura híbrida, combinando as abordagens mediador e agentes autónomos. A arquitectura AARIA (*Autonomous Agents at Rock Island Arsenal*) fornece uma funcionalidade integrada do MRP (*Manufacturing Resources Planning*) e MES (*Manufacturing Execution Systems*), numa abordagem hierárquica baseada em agentes, permitindo o diálogo com os clientes e os fornecedores, de forma a otimizar o escalonamento e reagir aos distúrbios que ocorram na fábrica [15]. Os recursos de fabrico são encapsulados como agentes autónomos e esta arquitectura utiliza um mecanismo avançado de escalonamento.

A abordagem MASCADA é constituída por agentes autónomos e inteligentes focando os sistemas de execução de fabrico [16].

A arquitectura HOLOS suporta o escalonamento dinâmico baseado em sistemas multi-agentes e a integração com sistemas legados. Esta abordagem foi estendida posteriormente para sistemas de fabrico multi-site e empresas virtuais [17]. O sistema DEDEMAS (*Decentralised Decision-Making and Scheduling*) tem por objectivo fornecer mecanismos para tomadas de decisão e escalonamento descentralizados, cobrindo os casos das operações *multi-site* de uma empresa, gestão da cadeia produtiva e empresas virtuais [18]. Os mecanismos de tomada de decisão são baseados numa extensão do protocolo *Contract Net*.

4. ARQUITECTURA BASEADA EM AGENTES

Na secção 2 foram apresentados os requisitos dos sistemas de fabrico emergentes e na secção 3 foram analisados alguns trabalhos de investigação na área das arquitecturas de controlo para sistemas de fabrico, que cumpram os requisitos analisados na secção 2. Apesar da contribuição desses trabalhos de investigação, alguns assuntos permanecem por solucionar e as arquitecturas apresentadas não solucionam simultaneamente os requisitos referidos, tais como o suporte de ambientes distribuídos e cooperativos, uma capacidade eficiente de auto-organização, uma fácil e rápida reacção e adaptação às mudanças, integração de humanos e integração com dispositivos físicos.

Adicionalmente, actualmente as arquitecturas de controlo focam a fase operacional do ciclo de vida e envolvem a capacidade de adaptação às mudanças. No entanto, pouca atenção é dada às mudanças do próprio sistema de fabrico, que a planta fabril sofre de forma continua, ao longo do seu ciclo de vida. As abordagens aos sistemas de fabrico distribuído devem considerar as mudanças impostas pelo ambiente no qual o sistema de fabrico actua e a mudanças do sistema de fabrico

durante a fase de reengenharia. Para atingir este requisito, é definido um novo nível de controlo que engloba as fases operacional e reengenharia do ciclo de vida de uma organização [19].

A arquitectura proposta para o desenvolvimento de aplicações de fabrico distribuídas suporta a nova abordagem ao controlo e tenta cumprir simultaneamente, de forma eficaz, os requisitos apresentados na secção 2.

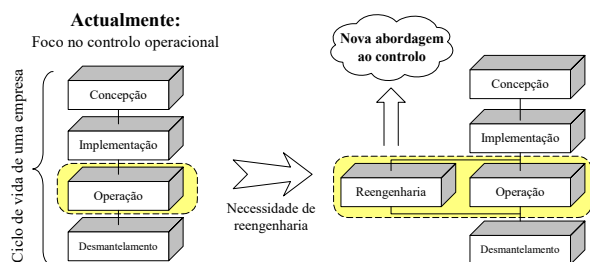


Figura 2 – Nova abordagem ao controlo

Os sistemas multi-agente são apropriados para ambientes de fabrico distribuído, uma vez que este tipo de aplicações apresenta características em que os agentes são eficazes para as solucionar, tais como modularidade, descentralização, extensibilidade e complexidade [20]. Analisando os benefícios da tecnologia de multi-agente é possível verificar que cumprem os requisitos na secção 2:

- **Autonomia (e modularidade)** - Um agente pode funcionar sem a intervenção directa de entidades externas, tais como humanos, e possui controlo sobre o seu comportamento.
- **Cooperação** - Os agentes interagem com outros agentes, através de uma linguagem de comunicação, de forma a cooperarem para atingir um objectivo comum.
- **Reactividade e Proactividade** - Os agentes percebem o seu ambiente e respondem rapidamente às mudanças que ocorrem. Por outro lado, os agentes não funcionam simplesmente em função de reacção ao ambiente, mas são capazes de tomar iniciativa, controlando o seu comportamento.
- **Adaptativa e descentralizado** - Os agentes podem organizar-se numa estrutura descentralizada e facilmente podem reorganizar-se em diferentes estruturas organizativas.

Como conclusão, os sistemas multi-agente são uma solução apropriada para implementar aplicações de controlo para sistemas de fabrico distribuídos que cumpram as especificações referidas na secção 2.

4.1 Conceitos básicos

A arquitectura proposta para sistemas de fabrico ágeis e cooperativos é constituída por um conjunto de agentes autónomos, cooperativos e inteligentes, formando uma plataforma de multi-agentes. Esta arquitectura

implementa alguns conceitos associados aos HMS, tais como a possibilidade de representar humanos, permitir diferentes estruturas organizativas e um holon possuir uma componente física e lógica, assim como algumas ideias dos BMS, tais como o papel de supervisor e a evolução dinâmica do sistema. Os agentes podem organizar-se em diferentes estruturas organizativas (auto-organização), de forma a otimizar os objectivos individuais e da comunidade, e combinam a robustez apresentada pelas arquitecturas hierárquicas e a reacção a distúrbios apresentados pelas arquitecturas heterarquicas.

A arquitectura é suportada por várias classes de agentes, para a representação dos componentes de uma empresa ou planta fabril: agentes operacional, supervisor, produto, tarefa e interface. O agente operacional representa os recursos físicos da planta fabril e o agente supervisor tem por objectivo o controlo de vários agentes operacionais, dependendo da estrutura organizativa. O agente interface permite a interacção do sistema com sistemas externos, sistemas legados, administração de sistema, etc.

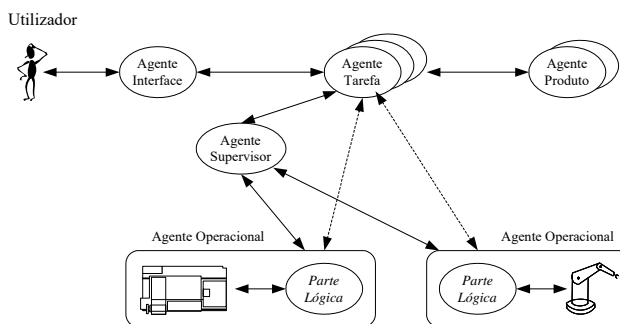


Figura 3 – Classes de agentes na arquitectura

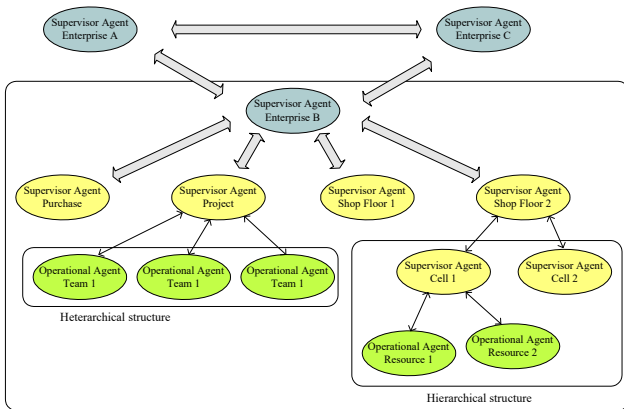
O agente produto representa o conhecimento do produto e a informação associada, tal como planos de processo, lista de material (*Bill of Materials*) e lista de recursos (informação estática). O agente tarefa é lançado para representar a execução de uma tarefa de forma a produzir o produto. Este agente possui a informação dinâmica acerca das ordens que são executadas pelo sistema de fabrico.

4.2 Estruturas organizativas

Uma característica importante da arquitectura proposta é a sua capacidade de auto-organização, que permite a reorganização dos agentes em diferentes estruturas organizativas, com especial atenção à estrutura completamente heterarquica (abordagem de agentes autónomos), que permite uma boa e rápida reacção a distúrbios, e outra arquitectura mais hierárquica (abordagem de federação ou mediador), que permite uma boa performance do sistema. No entanto, é possível adicionar dinamicamente novas estruturas organizativas.

A figura 4 representa a cooperação de várias empresas num ambiente distribuído, tal como as empresas estendidas, e um foco especial no sistema de fabrico de

uma empresa. Neste sistema de fabrico existem sub-



sistemas com diferentes estruturas organizativas.

Figura 4 – Estruturas organizativas na arquitectura

4.3 Mecanismos de Comunicação

A interacção entre agentes é assíncrona, i.e., um agente que envia a mensagem continua a sua execução sem a necessidade de esperar pela resposta. A arquitectura suporta vários modos de comunicação, dependendo da intenção e do conhecimento do agente: ponto-a-ponto (comunicação directa entre dois agentes), *multicast* (comunicação entre um agente e um grupo seleccionado de agentes) e *broadcast* (comunicação entre um agente e todos os agentes do sistema).

A linguagem de comunicação é baseada e derivada da linguagem KQML (*Knowledge Query and Manipulation Language*) e o formato das mensagens para a troca de dados usa a linguagem XML. De forma a standardizar os conteúdos das mensagens serão utilizadas ontologias. Os mecanismos de comunicação usam a plataforma TCP/IP (assim como *Fieldbus* para a comunicação com dispositivos físicos) e a arquitectura CORBA (*Common Object Request Broker Architecture*).

4.4 Gestão de Erros

Em ambientes fabris existem distúrbios que desviam o processo de fabrico do plano original, tais como avarias nas máquinas, falta de material, novas encomendas e modificações nas especificações da encomenda. Adicionalmente, é importante considerar o processo de reengenharia da planta fabril, que implica mudanças na sua configuração, tais como adicionar ou remover máquinas.

Em caso de distúrbios, o sistema deverá responder dinamicamente e rapidamente, usando mecanismos de acordo com o tipo de distúrbio: introdução de novas encomendas (re-escalonamento das ordens de fabrico), modificação das ordens (libertar todas as ordens de fabrico já escalonadas para os agentes operacionais e reescalonar novas ordens de fabrico), atraso numa tarefa (re-escalonamento das próximas tarefas afectadas pelo atraso), avaria na máquina com destruição do produto (re-escalonamento de todas as ordens de

fabrico relacionadas com o produto) e avaria na máquina sem a destruição do produto (re-escalonamento apenas das ordens de fabrico incompletas). No caso de avaria de uma máquina é necessário considerar o estado da máquina após a falha: se a máquina está operacional, esta pode ser utilizada no processo de re-escalonamento, caso contrário, é necessário re-escalonar todas as ordens de fabrico alocadas à máquina.

4.5 Aprendizagem

A arquitectura apresentada deve ser capaz de se adaptar à volatilidade do ambiente e a aprendizagem é um aspecto fundamental para que os agentes tratem este contexto emergente e incrementem a sua performance. Existem situações nas quais os agentes devem iniciar o processo de aprendizagem, tais como: mudança na configuração do sistema, ocorrência de falhas e finalização de tarefas com sucesso.

O processo de aprendizagem pode ser realizado de duas formas distintas: auto-aprendizagem, sem a intervenção externa, e treino, com a ajuda de tutores (especialistas). Nesta arquitectura o processo de aprendizagem utiliza principalmente mecanismos de auto-aprendizagem, baseado na informação histórica, e mecanismos de treino. Este último modo de aprendizagem é implementado através do uso dos agentes interface, que permitem que especialistas introduzam procedimentos e regras que regulem o comportamento do agente.

5. ARQUITECTURA PARA UM AGENTE GENÉRICO

A arquitectura para um agente genérico, que pertence à arquitectura apresentada no ponto anterior, é baseada em quatro módulos e numa base de dados de conhecimento, que contém a informação relevante para o agente. Estes módulos interagem entre si, e dois deles interagem com o ambiente externo.

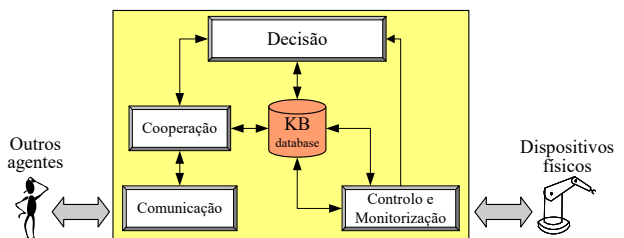


Figura 5 – Arquitectura para um agente genérico

5.1 Módulo de decisão

O módulo de decisão é o cérebro do agente e controla as actividades e o fluxo de informação no interior do agente. Este módulo envolve, entre outros, a resolução de problemas, tomadas de decisão para escalonamento, planeamento e alocação de ordens. O processo de tomada de decisão envolve a análise da informação disponível, a aplicação dos mecanismos de tomada de decisão, a avaliação da decisão e aprendizagem (se

necessário). A aprendizagem e avaliação da decisão deve considerar os resultados da decisão tomada.

Para suportar o processo de tomada de decisão, o módulo usa o conhecimento e informação armazenada na base de dados de conhecimento, e quando a informação disponível não é suficiente para tomar a decisão, é iniciado um processo de cooperação de forma a encontrar a informação necessária para tomar a decisão. Adicionalmente, o agente é inteligente e as decisões podem ser tomadas baseadas no conhecimento adquirido nas experiências anteriores.

5.2 Módulo de cooperação

Este módulo tem como objectivo a gestão da cooperação com outros agentes, através da requisição de cooperação, a compilação das respostas e o envio das respostas ao módulo de decisão.

Este módulo define um protocolo de negociação baseado no bem conhecido *Contract Net Protocol* [21], estendendo as funcionalidades originais e incluindo as seguintes características:

- O processo de negociação deve permitir múltiplas iterações, de forma a ter replaneamento dinâmico e otimizado.
- Diferentes formas de especificar o anúncio de tarefas e propostas, de forma a suportar a apresentação de propostas com a especificação de quantidades parciais da tarefa.

O módulo de cooperação possui inteligência suficiente para replicar uma mensagem, no caso da mensagem não ser adequada ao pedido inicial. As múltiplas iterações durante o processo de negociação são o resultado de múltiplos pedidos de cooperação.

5.3 Módulo de comunicação

O módulo de comunicação envolve a necessidade de standardizar a interacção entre agentes distribuídos e define o tipo de comunicação (conexão, meio e intenção) e a linguagem de comunicação. Um aspecto adicional é a construção de ontologias que definam o vocabulário que é utilizado na comunicação entre agentes, assim como o conhecimento relacionado com esses termos. O objectivo das ontologias é criar entendimento entre entidades cooperativas, permitindo a troca de conhecimento entre agentes e a capacidade de reutilizar esse conhecimento [22].

O módulo de comunicação é dividido em três níveis: conteúdos, mensagem e físico. Quando o módulo de cooperação pretende enviar uma mensagem para outro agente, envia para o módulo de comunicação a informação que pretende enviar e a indicação dos agentes para onde deve ser enviada. O nível de conteúdos interpreta a informação e aplica a ontologia apropriada, de forma a standardizar a troca de dados. O nível de mensagem formata a mensagem a enviar, usando uma linguagem derivada do KQML. Este nível lida também com os parâmetros de comunicação, tais como os agentes destino e os seus endereços. O nível

físico permite a iteração física para o envio e recepção das mensagens, utilizando a arquitectura CORBA.

5.4 Módulo de controlo e monitorização

Este módulo tem por objectivo o controlo e monitorização da execução operacional do agente. Na parte de controlo, são realizadas as funções de despacho e gestão, enquanto na parte de monitorização, é realizada a monitorização das actividades executadas pelo dispositivo físico associado ao agente.

Este módulo é dividido em quatro sub-módulos: gestão, despacho, monitor e controlo operacional. O sub-módulo de gestão pretende gerir a execução da tarefa, envolvendo várias operações, tais como apagar um programa na máquina, descarregar um programa para a máquina, iniciar a sua execução e esperar pelo fim da execução do programa.

O sub-módulo de despacho formata e despacha as tarefas para os dispositivos físicos. O sub-módulo monitor permite a monitorização activa e passiva. A monitorização passiva não envolve mecanismos de detecção de eventos, enquanto que a monitorização activa está relacionada com a detecção da ocorrência de alarmes e implementa mecanismos de detecção de eventos baseados na notificação da ocorrência de alarmes subscritos por agentes distribuídos [17]. O sub-módulo de controlo operacional permite a comunicação com os dispositivos físicos ou com os sistemas legados, para a execução de tarefas atribuídas ao agente local. A comunicação será realizada utilizando interfaces reutilizáveis através do uso da arquitectura CORBA.

5.5 Base de dados de conhecimento

A base de dados de conhecimento local armazena todo o conhecimento relacionado com o comportamento do agente e da comunidade à qual o agente pertence. A informação armazenada envolve diversos tipos de conhecimento, tais como objectivos, procedimentos, regras, restrições, técnicas e estruturas organizativas e experiência.

A abordagem à base de dados tem duas dimensões: informação individual - global e informação genérica - específica. A informação individual é relacionada com a informação local do agente, enquanto a informação global é relacionada com a informação do sistema como um todo. A parte genérica da informação é relacionada com a informação comum a todos os agentes, enquanto que a parte específica é relacionada com a informação específica a cada agente.

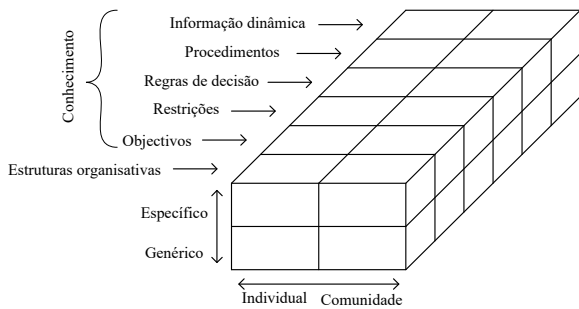


Figura 6 – Base de dados de conhecimento multi-dimensional

A informação dinâmica representa diversos tipos de informação, tais como o estado actual do agente, a capacidade de execução, o plano de trabalho, o *setup* actual, etc.

5.6 Customização da arquitectura genérica para um agente

Todos os tipos de agentes que pertencem à arquitectura proposta são baseados na arquitectura de um agente genérico, descrita no ponto anterior. Por exemplo, o agente operacional usa todos os módulos definidos na arquitectura, customizando-os de acordo com as suas especificidades.

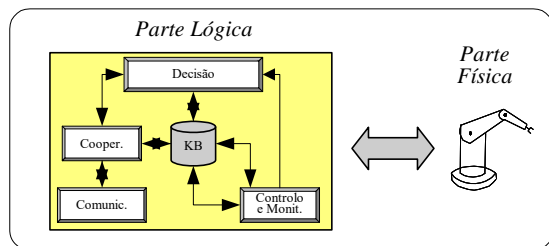


Figura 7 – Agente operacional

Este tipo de agente usa uma característica importante dos HMS: um holon pode possuir uma parte lógica (componente de *software*) e uma parte física. Desta forma, a parte lógica controla o recurso físico através de um sistema de comunicação apropriado e definido no módulo de controlo e monitorização local. Utilizando outro conceito dos HMS, na qual um holon pode ser parte de outro holon, é possível modelizar um agente operacional que represente uma célula de fabrico, constituída por um conjunto de vários agentes, cada um dos quais representando um recurso da célula de fabrico.

No entanto, existem agentes que não utilizam todos os módulos definidos na arquitectura de um agente genérico. Um exemplo, é o agente produto que não utiliza o módulo de controlo e monitorização local, uma vez que este tipo de agente não interage com dispositivos físicos.

6. CONCLUSÕES

Este artigo reflecte a investigação realizada nos sistemas de fabrico distribuídos e cooperativos, e o esforço no desenvolvimento de uma arquitectura que

cumpra os principais requisitos apresentados por este tipo de sistemas, como sejam, a distribuição geográfica de aplicações, a cooperação, a auto-organização e a integração de humanos e sistemas legados.

Neste artigo é apresentada uma arquitectura baseada em agentes para o desenvolvimento de aplicações de controlo de sistemas distribuídos que suporta uma nova abordagem de controlo, que inclui a fase de reengenharia, que implementa alguns conceitos associados aos sistemas de fabrico holónico e biónico e utiliza sistemas multi-agentes como plataforma. Adicionalmente, é apresentada a arquitectura de um agente genérico, sendo analisados os seus vários componentes.

Este é um trabalho que está em curso e apenas os resultados preliminares são apresentados.

7. REFERÊNCIAS

- [1] Leitão, P. and F. Restivo (2000). A framework for Distributed Manufacturing Applications. In: *Proceedings of ASI'2000 International Conference*, Bordeaux, France, September 2000.
- [2] Groover, M. P. (1987). *Automation, Production Systems and CIM*. Prentice-Hall.
- [3] Rembold, U. and B.O. Nnaji, (1993). *Computer Integrated Manufacturing and Engineering*. Addison-Wesley.
- [4] Camarinha-Matos, L. M. and H. Afsarmanesh (1999). Infrastructures for Virtual Enterprises: a summary of achievements. In: *Proceedings of PRO-VE'99 - IFIP Int. Conf. On Infrastructures for Virtual Enterprises* (L.M. Camarinha-Matos, H. Afsarmanesh - Eds.), Kluwer Academic Publishers, ISBN 0-7923-8639-6, pp. 483-490, Porto, Portugal.
- [5] Shen, W. and D. Norrie (1999) Agent-based systems for intelligent manufacturing: A state-of-the-art Survey. *Knowledge and information systems: an International Journal*, Vol1, nº 2, pp 129-156.
- [6] Bauer, A., R. Bowden, J. Browne, J. Duggan and G. Lyons (1991). *Shop Floor Control Systems - From Design to Implementation*. Chapman & Hall.
- [7] Gullander, P. (1999). *On Reference Architectures for Development of Flexible Cell Control Systems*. PhD thesis, Gotenborg University, Sweden.
- [8] Leitão, P. and A. Quintas (1997). A Manufacturing Cell Controller Architecture. In: *Proceedings of Flexible Automation and Intelligent Manufacturing Conference*, Begell House Inc, ISBN:1-56700-089-4, Edited by William G. Sullivan and M. Munir Ahmad, Middlesbrough, June, pp 483-493.
- [9] Teunis, G., P. Leitão and M. Madden (1998). A New Architecture for Flexible Shop Control Systems. In: *Proceedings of Integration in Manufacturing Conference*, Gotengorg, 6-8 October, pp 699-709.

- [10] Warneke, H.J. (1993). *The Fractal Company*. Springer-Verlag.
- [11] Okino, N. (1993). *Bionic Manufacturing System*. In J. Peklenik (ed.) *CIRP, Flexible Manufacturing Systems Past-Present-Future*, pp. 73-95.
- [12] Bongaerts, L. (1998). *Integration of Scheduling and Control in Holonic Manufacturing Systems*. PhD thesis, Katholieke Universiteit Leuven, Belgium.
- [13] Tharumarajah, A., A. Wells, L. Nemes (1996). *Comparison of the Bionic, fractal and holonic manufacturing systems concepts*. In *International Journal of Computer Integrated Manufacturing*, Vol.9, n°3, pp 217-226.
- [14] Maturana, F. and D. Norrie (1996). Multi-Agent Mediator Architecture for Distributed Manufacturing. *Journal of Intellivent Manufacturing*, Vol. 7, pp. 257-270.
- [15] Parunak, H. Van Dyke, A. Baker and S. Clark (1998). The AARIA agent architecture: from manufacturing requirements to agent-based system design. In: *Workshop on Agent-based Manufacturing*, ICAA'98, Minneapolis, 10 May.
- [16] Heikkilä, T., M. Kollingbaum, P. Valckenaers and G.-J. Bluemink (1999). manAge: An agent architecture for manufacturing control. In: *Proceedings of the 2nd International Workshop on Intelligent Manufacturing Systems*, KULeuven, pp.127-136.
- [17] Rabelo, R. and L.M. Camarinha-Matos (1994). Multi-Agent based Dynamic Scheduling. *Int. Journal on Robotics and Computer Integrated Manufacturing*, Vol II, n.4, pp 303-310.
- [18] Tönshoff, H.-K., I. Seilonen, G. Teunis and P. Leitão (2000). A Mediator-based approach for decentralised production planning, scheduling and monitoring. In: *Proceedings of CIRP International Seminar on Intelligent Computation in Manufacturing Engineering*, 21-23 June, Capri, Italy.
- [19] Leitão, P., J. Barata, L.M. Camarinha-Matos, R. Boissier (2001). Trends in Agile and Co-operative Manufacturing. Submitted to the Low Cost Automation 2001, Dusseldorf, Germany, 8-10 October.
- [20] Parunak, H. Van Dyke (1998). *What can Agents do in Industry, and Why? An Overview of Industrially-Oriented R&D at CEC*. Industrial Technology Institute; CIA'98.
- [21] Smith, R.G. (1980). *The Contract Net Protocol: High-Level Communication and Control in a Distributed Solver*. *IEEE Transactions on Computers*, Vol. C-29, N°12, pp 1104-1113.
- [22] Gruber, T. (1993). A translation approach to portable ontologies. *Knowledge acquisition*, 5 (2), pp 199-220.